



Xi, W., Munro, ATD., & Barton, MH. (2008). Architecture of achieving QoS for multiple flows per node in WLANs. In *IEEE International Workshop on Quality of Service (IWQoS), Enschede, Netherlands* (pp. 191 - 198). Institute of Electrical and Electronics Engineers (IEEE).
<https://doi.org/10.1109/IWQOS.2008.27>

Peer reviewed version

Link to published version (if available):
[10.1109/IWQOS.2008.27](https://doi.org/10.1109/IWQOS.2008.27)

[Link to publication record in Explore Bristol Research](#)
PDF-document

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

Architecture of Achieving QoS for Multiple Flows per Node in WLANs

Weihua Helen Xi, Alistair Munro, Michael Barton

Department of Electrical Electronic Engineering, University of Bristol, Bristol, UK

Email: {Helen.Xi, Alistair.Munro, M.H.Barton}@bristol.ac.uk

Abstract—This paper proposes an effective architecture to achieve QoS in distributed WLANs when multiple traffic streams are present and flowing through every single station. The 802.11e standard specifies a multiple-state-machine (MSM) structure in the Medium Access Control (MAC) to address this QoS concern. However, our research indicates that the MSM structure was suboptimal. It increases the collision rate and weakens the stability of the network. Instead, we propose a QoS scheduler over the MAC state machine (called Local Scheduler Multi Flow, LSMF): the proposed LSMF architecture prevents every flow in a node from initiating its own contention simultaneously, consequently, improves the overall network throughput and stability. We evaluate the performance of our LSMF model through mathematical analysis and simulations in comparison with the 802.11e MSM model, and prove the LSMF architecture is superior in both QoS effectiveness and flexibility for multiple flows per node case.

I. INTRODUCTION

The evolution of Wireless Local Area Network (WLAN) technologies aims to maximise the utilisation of the wireless medium. This includes faster data delivery at the physical layer (PHY) and effective management of the distributed resource allocation and contention decisions at the Medium Access Control (MAC) layer. The increased over-the-air data speed encourages end users to run multiple applications simultaneously on a single device. A key function that the MAC can implement in this situation is to match the arrival characteristics of traffic flows input to a device on one link with the resource available on their destined output link with Quality of Service (QoS).

QoS in packet networks (called ‘packet QoS’ below) depends on a scheduler that observes the service requirements of packet flows. It has been the subject of much research, e.g. Weighted Fair Queuing (WFQ) [6]. This is an approach that applies to a single forwarding device and it does not survive distribution to systems such as the distributed WLAN, which is our main concern in this paper.

The Enhanced Distributed Coordination Function (EDCF) [1] is not a packet QoS model in the sense described above. It discriminates service for a limited number of access categories (ACs), each of which contends for access to the medium with other QoS-capable devices (QSTAs) that have traffic. It is a distributed algorithm offering priority-based service differentiation to flows in the same access category (AC) across all nodes in the system. At link access level, it allows QSTAs with traffic in the high priority ACs to resolve contention ahead of traffic in lower priority ACs, with fairness and throughput

characteristics limited by the constraints of the CSMA/CA MAC of 802.11.

The question that we address in this paper is: does EDCF deliver the benefits in performance that it claims, by comparison with packet QoS applied to 802.11; and can it be improved?

The problem that we perceive with EDCF is that it associates QoS with priority. There is indeed a priority aspect to packet QoS: when the scheduler has several packets to choose from, all of which are eligible for transmission, some of which are more urgent than others, then the packets must be ranked in importance with respect to maintaining QoS overall. The ranking defines the priority, which will change as the flow populations change. However, the distributed contention resolution process must still take place between QSTAs to ensure that the urgent packets gain access. In addition, all QSTAs must operate their QoS policies using the same rules, as EDCF does for priority ACs.

Recent research, e.g. [7] [13], has aimed to support packet QoS in the distributed WLAN. Most packet QoS research does not elaborate much on the multi-flow per node case. Analysis and simulations are based on the single-flow assumption. 802.11e articulates how multiple flows are coordinated in one QSTA. However, in most research papers, such as [2] [3] [4] on 802.11e, for the sake of simplification, each QSTA transmits only one traffic class. Even most analytical models, e.g. [5], are based on the assumption that each station serves only one AC. This simplifies the investigation but ignores the influence of internal contentions on the whole system. The effect becomes significant as traffic load increases. For example, we demonstrate that the internal contention in 802.11e reduces medium utilisation. Every AC has its own state machine acting as an independent node. The AC backoff timers are likely to converge while contending for a transmission. As this happens, the collision probability between QSTAs becomes greater. A mathematical model is used to prove this.

With the faster delivery capacity of the PHY, a single device is better capable of supporting multiple users and multiple applications simultaneously. Our proposal aims to provide an improved basis for EDCF, and to provide a more comprehensive analysis that could be used to verify packet QoS algorithms in more general scenarios. However, it must be proved that our alternative proposal can be operated as a distributed algorithm.

The essential element of our proposal is that each packet is processed by a queuing scheduler before it goes to the MAC

state machine. When a packet is presented to the MAC, it is known to be the most urgent packet local to that QSTA (how this is done is beyond the scope of this paper). However it must still compete with other QSTAs to assert this urgency system-wide. In line with EDCF and adaptations of WFQ, its initial contention window (CW) slot time will be selected according to this urgency.

The paper is organised as follows: Section II gives a brief introduction of Packet QoS and research scope. Section III describes the LSMF model. Section IV presents the simulation results. Section V gives a further discussion about the benefits of our model. Section VI concludes the paper.

II. PACKET QOS

In packet QoS, the processing of inbound traffic by a forwarding device capable of differentiating service and delivering QoS is done in four steps:

- Packets arrive and are classified according to some local criteria into flows, and an interface on which they will be forwarded (next hop) is selected;
- Flows will be metered over some time period to check that they comply with their claimed arrival patterns, e.g. average rate, peak rate, burst duration. They will be marked accordingly with reference to the metering period, e.g. green for at or below average, yellow for above average but below peak and within burst period, red for any that exceed these limits;
- Marked packets are queued according to the service they require, which typically includes a (minimum) output bit rate, and possibly, delay and delay variation;
- A scheduler examines the population of packets in the queue, or queues, for each link and makes a selection of the next packet to transmit according to the mix of service requirements, the current demands of the population given the resource that is available, and the marking of eligible packets. The algorithms for making the selection are varied, and are described in more detail below.

The scheduler is the key element in this process. At low loads (i.e. the traffic offered to an output interface), it should be essentially work-conserving, i.e. whenever a packet arrives, it will be transmitted immediately, incurring only the delay of the local datapath. In this situation, the packet QoS has no effect. Flows may, however, be rate-limited by discarding excess traffic (the red packets). As load increases, the scheduler becomes increasingly less work-conserving and will have to police and shape flows to ensure the service requirements are met.

Although it should be possible to configure the device in such a way that the traffic specifications and service requirements are consistent, reality undermines this in two ways:

- Flows may be bursty, either inherently, or due to upstream aggregation in which phasing of components of the flow, or earlier packet QoS shaping, or simply accumulated processing delay, distorts an initially uniform input rate;
- Packets have varying lengths. IP packet flows from a device contain large numbers of packets of around 64

bytes, 100 bytes and 1500 bytes, representing ARP, DNS and TCP/HTTP protocols (the data, not control traffic) respectively. VoIP packets are all the same size and rate at source for individual calls, from 40 to 240 bytes according to codec. MPEG encoded video packets vary widely in size although the source rate is uniform. A typical minimum is 188 bytes, and maximum size may exceed 1500.

These issues must be taken into account when analysing performance.

We choose to concentrate our research scope on distributed WLANs, more specifically:

- Our model is restricted to the single-hop mesh case where all stations are able to communicate directly with each other, i.e. no hidden nodes.
- Each station has several traffic types to transmit and the destination stations can be any of the nodes in this network. Every station has an equal level of service priority. The level of QoS depends on the traffic type itself, and is not affected by its source or destination.
- Successful communication among stations in a network does not depend on PHY transmission, whose impact on system performance beyond the scope of this paper.

Certain terms we use especially for the multi-flow case are explained below:

- *Node/station/QSTA/user* - we use these terms interchangeably in this paper.
- *Flow/queue/stream* - a flow is a traffic stream with the same content type generated from the same node. 802.11e puts the same traffic type in a queue, also called AC.

III. LOCAL SCHEDULER FOR MULTI-FLOW (LSMF)

A. Introduction to LSMF

In 802.11e, each AC initiates an EDCF state machine running in the MAC to contend for the radio channel, as illustrated in Fig. 1. So 802.11e has a Multiple-State-Machine (MSM) structure to support multiple flows. 802.11e defines four ACs representing different traffic types. The fact that there are, in effect, four times as many packets contending for the channel increases the collision probability, which leads to more retransmissions.

We propose a new structure, a Local queuing Scheduler for Multi-Flow per node (LSMF), replacing the MSM mechanism with a queuing scheduler above the MAC and without losing the ability to support QoS. The queuing scheduler, shown in Fig. 2 (we use QSTA to mean QoS station in 802.11e and Q'STA to denote the LSMF), manages four queues with each one representing one AC. The scheduler extracts the packet from the queue according to a packet QoS algorithm and sends the packet to the MAC layer. The MAC layer only has one state machine contending for the radio channel with the specific AIFS[AC] and CW[AC] corresponding to the packet's AC. The QoS assurances scheme in [14] has a similar node structure and further interesting features will be explained at the end of this chapter.

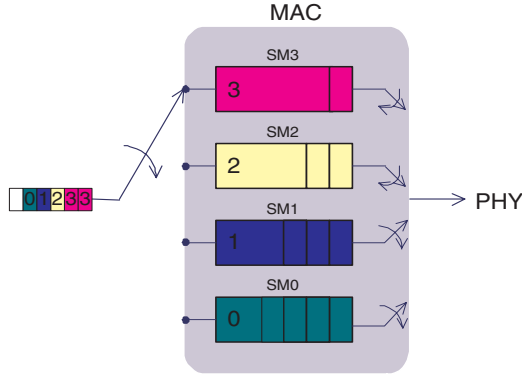


Fig. 1. 802.11e QSTA - there is a state machine (SM) for each traffic class

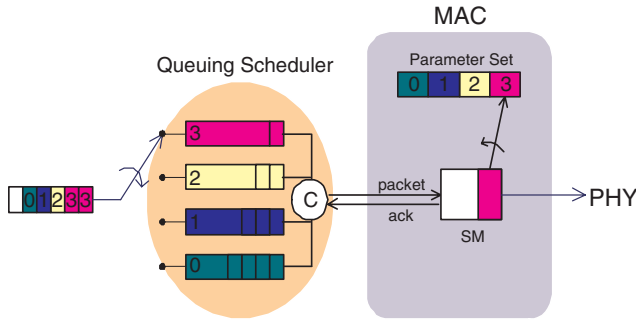


Fig. 2. Q'STA in LSMF

B. Algorithm in the queuing scheduler

The queuing scheduler above the MAC in the Q'STA selects a traffic queue to send a packet to the MAC. The algorithm can be any of the classical QoS schemes, e.g. round-robin, weighted fair scheduling (WFQ), priority scheduling [15].

In this model, the controller ('C') allocates a weight to each flow. The one with the smallest weight is sent to the MAC. The weight of the flow is the sum of its AIFS and its backoff time (we refer to this as the 'waiting time') described in the MAC contention procedure. The initial weight of each flow consists of AIFS[AC] and a slot time multiplied by a random integer drawn from a uniform distribution over [0, CW[AC]]. When the packet in the MAC is successfully received or dropped through exceeding the retransmission limit, an acknowledgement will be sent to inform the scheduler that the MAC is ready to process the next packet. The weights of the four flows will be updated after receiving the acknowledgement. The updating algorithm works as in the CSMA/CA procedure. The immediate past flow winner, which had the smallest weight, is reset to draw a new random integer over [0, CWmin[AC]]. The other three deferred flows' backoff time will be their previous weight subtracted by the time passed, which was the winner's weight. Implementation detail is given in the next section. After updating, the flow with the smallest weight will send its head packet to the MAC as the new internal winner to contend with other Q'STAs in the network.

There is a regular scan for packets of the highest priority:

when the packet in the MAC fails a transmission, the system should scan quickly to see if there is any delay-sensitive packet in the queue. If so, this packet will come into the MAC and prepare for transmission instead of the original lower priority packet. Without this facility, the highest priority traffic might suffer from longer delay if low priority packets are allowed continuous retransmission attempts. Under this heuristic policy, low priority packets will not take control of the MAC for too long and jeopardise the highest priority traffic.

C. Analysis of the collision probability

We now compare the collision probabilities in 802.11e and our alternative model. Figures describing the CSMA/CA process using a discrete-time Markov chain can be found in many papers, e.g. [8] [9]. Although there is some discrepancy in the Markov model given in these papers, the effect on the throughput is small. The stationary probability distributions can be computed using Markov theory and balance equations, which are standard techniques. We define the probability τ_i ($i = 0, 1, 2, 3$, representing AC_BK, AC_BE, AC_VI and AC_VO) that an AC i initiates a transmission at a generic timeslot during the backoff period. The probability that the channel is busy during the backoff period is expressed by p . p_c is the probability that a collision happens. w_j is the CW size at the retry stage j (retry limit is m). The derivation of the following Equation 1 can be found in the cited papers.

$$\tau_i = \frac{\frac{1-p_{c,i}^{m+1}}{1-p_{c,i}}}{\frac{1-p_{c,i}^{m+1}}{1-p_{c,i}} + \sum_{j=0}^m \frac{p_{c,i}^j w_j}{2(1-p_i)}} \quad (1)$$

This can be applied to any AC. To make the comparison easier, we assume only two traffic types (AC_VO and AC_VI) in each station. The probability τ that a QSTA transmits is that at least one AC transmits.

$$\tau = 1 - (1 - \tau_3)(1 - \tau_2) \quad (2)$$

Assume the network has N QSTAs. Each QSTA has two ACs and both of them have packets to send. The probability p_i that a particular type of AC i resumes its backoff timer because of the busy channel means that at least one other QSTA occupies the channel, or the other AC within the same QSTA does. So p_i can be expressed as:

$$p_3 = 1 - (1 - \tau)^{N-1}(1 - \tau_2) \quad (3)$$

$$p_2 = 1 - (1 - \tau)^{N-1}(1 - \tau_3) \quad (4)$$

When an AC's backoff timer reaches zero and it attempts to transmit, the probability $p_{c,i}$ that a collision occurred can be given as:

$$p_{c,3} = 1 - (1 - \tau)^{N-1} \quad (5)$$

$$p_{c,2} = 1 - (1 - \tau)^{N-1}(1 - \tau_3) \quad (6)$$

To make the mathematical model approximate a Markov chain, CWmin and CWmax of AC_VO are set to be 15 and 127, AC_VI to be 31 and 255. The retry limit is 4. From Equation 1 - 6, the probability τ_3 for AC_VO and τ_2 for AC_VI can be expressed as:

$$\tau_3 = \frac{2(1 - \tau_2)(1 - p_{c,3}^4)}{2(1 - \tau_2)(1 - p_{c,3}^4) + 15 + 31p_{c,3} + 63p_{c,3}^2 + 127p_{c,3}^3} \quad (7)$$

$$\tau_2 = \frac{2(1 - p_{c,2}^4)}{2(1 - p_{c,2}^4) + 31 + 63p_{c,2} + 127p_{c,2}^2 + 255p_{c,2}^3} \quad (8)$$

In the Q'STA, we assume the scheduler is using the pseudo-code MAC described the previous subsection, and the CW sizes are set as above. We define a parameter a :

$$a = \frac{\text{load}[AC_VO]}{\text{load}[all]}$$

Using the computer simulation for this scheduler, the controller chooses AC_VO 2.06 times as often as AC_VI. So a is $2.06/(1+2.06)$. There is only one state machine in each Q'STA, so the τ' depends on which traffic type the Q'STA is serving:

$$\tau' = \{\tau'_3, \tau'_2\}$$

We assume at a generic timeslot, the number of other Q'STAs serving AC_VO is $a \times (N-1)$. Thus, the busy channel probability for any type of AC is:

$$p' = 1 - (1 - \tau'_3)^{a(N-1)}(1 - \tau'_2)^{(1-a)(N-1)} \quad (9)$$

In the Q'STA, there is no internal collision, so $p'_c = p'$. Together with the Equation 1, τ'_3 for AC_VO' and τ'_2 for AC_VI' can be expressed as:

$$\tau'_3 = \frac{2(1 - p_c'^4)}{2(1 - p_c'^4) + 15 + 31p_c' + 63p_c'^2 + 127p_c'^3} \quad (10)$$

$$\tau'_2 = \frac{2(1 - p_c'^4)}{2(1 - p_c'^4) + 31 + 63p_c' + 127p_c'^2 + 255p_c'^3} \quad (11)$$

From Equation 1 - 11, we can compare the collision probabilities of the two models using numerical techniques and show the results in Figure 3.

The solid blue line is for the above LSMF model and it has a lower collision rate than the collision rates of AC_VO and AC_VI in the 802.11e model. When the value of a varies from [0, 1], the blue line will change accordingly. The more load from the AC_VO', the higher the blue line will become. When a is 1, which means the node has only chosen AC_VO', the collision rate will be at its highest value, as the dashed blue line shows, but it is still lower than those in the 802.11e model. The dash-dot line is when a is 0. The 802.11e QSTAs with multiple flows attempt to control medium access more aggressively, but this causes high collision rates.

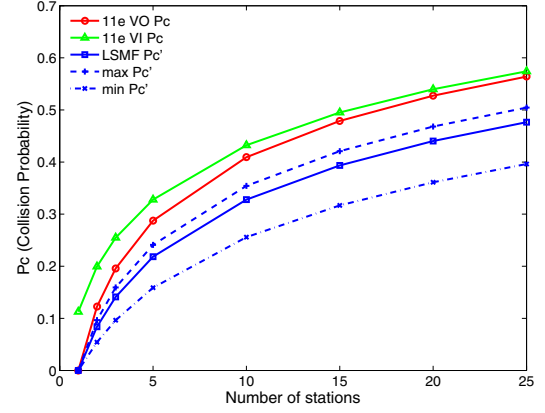


Fig. 3. The probability p_c that a station has a collision

IV. SIMULATIONS AND PERFORMANCE EVALUATION

A. Simulation Environments

The 802.11e model and the LSMF model were developed based on the OPNET [10] standard model to support multiple flows. We use VOIP, MPEG, TCP/HTTP traffic to represent different priorities AC_VO, AC_VI, AC_BE and AC_BK. Details of these traffic specifications are listed in Table I. Each

TABLE I
TRAFFIC SPECIFICATIONS

	Traffic arrival rate (bps)	Packet size (byte)	Interarrival rate
AC_VO	16K	80	40ms
AC_VI	4M	uniform[188, 1500]	1.688ms
AC_BE	100K	1500	120ms
AC_BK	100K	1500	120ms

station has all the four ACs. The network is loaded with a varied number of stations from underload to overload. The RTS/CTS mechanism is applied when the packet length is above 256 Bytes. This helps to reduce the time wasted due to collisions. The retry limit is 7 for short length packets and 4 for long length packets. The SIFS is $16\mu s$ and aSlotTime is $9\mu s$ (they are PHY dependent; 802.11a OFDM 54Mbps is applied here). Values of AIFS[AC] and CW[AC] are set with the default values defined in the 802.11e standard [1]. The time spent on the control frames, RTS and CTS (or ACK), is $28\mu s$ and $25.8\mu s$, respectively. The time spent on the overall extra bits (including preamble, headers and padding) is around $25.8\mu s$ to $28.1\mu s$.

As a measure of performance, we use throughput of successfully transmitted packets. The load on the network is the traffic generated for transmission. Dropped packets, i.e. those which have reached the retry limit and have been discarded, are also counted. All packets originated by the sources are recorded, i.e. the memory available for internal buffering is made unbounded, to allow clearer identification of potential instabilities. The delay is counted from the time the packet is generated until it is being received successfully.

The performance under different traffic load scenarios is the outcome of the investigation. To find out the maximum throughput at 54Mbps, we use only one station transmitting the AC_VI traffic (no collisions in this case). The throughput is about 19.5Mbps. With the traffic settings in Table I, 5 stations have the total load of 21.08Mbps. We consider the network with 5 stations is 100% loaded, with 6 stations is 120% loaded, etc.

B. Comparison of how effectively QoS is achieved

Simulations were run with 1 station, 2 stations,..., and 10 stations transmitting, which means the network was loaded from 20% to 200%, as shown in the x-axis in Figure 4. The y-axis represents the ratio of the packets successfully transmitted to the packets generated for transmission. When the network is underloaded, all the traffic flows can be transmitted and the packet in the buffer of each traffic type is no more than one. There is no any significant difference between the 802.11e model and the LSMF solution. When the network is overloaded, that is, when QoS is needed; the throughput achieved in the LSMF solution for every traffic type is always higher than in 802.11e. There are two possible reasons why the ratio of Throughput/Load becomes smaller when the system is loaded with more traffic. One reason is that more arriving packets are waiting in their queues. For example, the AC_VI packets are arriving for transmission very frequently and the buffer occupancy becomes larger. The lowest priority AC_BK has hardly had any chance to transmit and its buffer occupancy also becomes larger. So their ratios of sent packets to requested packets decreasing. Another reason is that dropped frames cause the ratio to become smaller when there are many contending entities. For example, the buffer occupancy for AC_VO is never more than 1, which means all its packets are not accumulated and are therefore transmitted on time without buffering, but its ratio is still less than 1. This is caused purely by the dropped frames.

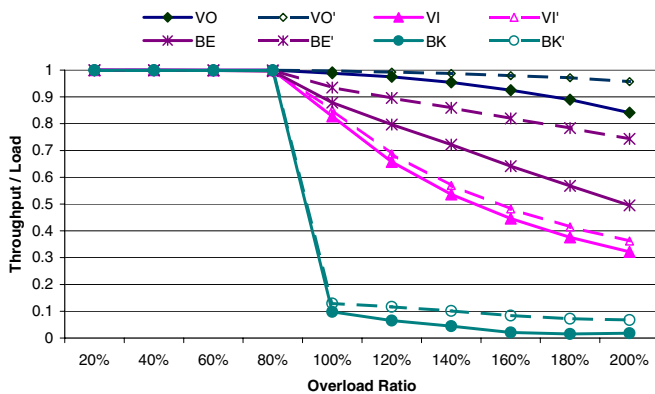


Fig. 4. Throughput/Load per AC in networks with different number of stations

The reason why 802.11e has lower throughput than the LSMF is its higher collision rate, which causes a high rate

of dropped packets. Figure 5¹ shows the ratio of dropped packets to successfully sent packets. For example, when the network has 10 QSTAs (the network is 200% loaded), for 100 successfully sent VOIP packets, the 802.11e has 18.7 VOIP packets dropped. That is why Figure 4 shows the ratio of throughput to load is 0.84 ($100/(100+18.7) = 0.84$), given there are no VOIP packets backlogged in the buffer. However, the LSMF solution only sacrifices 4.5 VOIP packets for every 100 that are successfully transmitted. In 802.11e, there are more transmission attempts, which lead to more collisions. Though the use of RTS/CTS helps to reduce the time wasted on collisions, a packet still has to be discarded when it reaches the retry limit². The high frame drop rate caused by collisions in 802.11e may jeopardise the quality of voice and video transmissions.

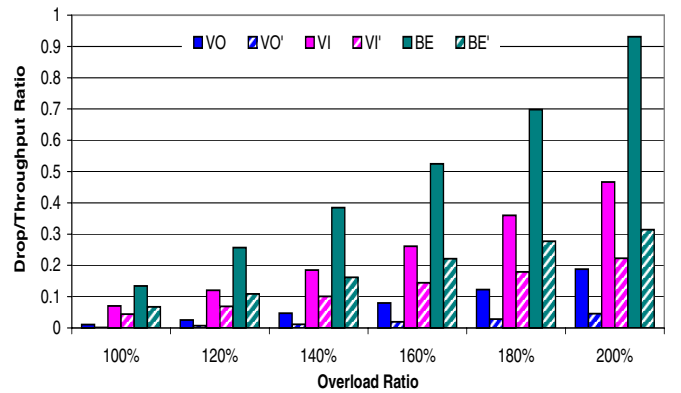


Fig. 5. Ratio of Dropped Packets over Successfully Sent Packets

These improvements in LSMF impact the delay. In the underloaded cases, the number of packets accumulated in the buffer is no more than one. There are fewer contention entities. This leads to very few collisions and retransmissions. The delay is under control. Table II shows the mean delay of each traffic type in both models. This delay does not include the time spent on transmission, i.e. it is purely the time waiting before transmission. All the delays³ in an underloaded network are small and do not show much variation.

TABLE II
MEAN DELAY IN AN UNDERLOADED NETWORK

	20%		40%		60%		80%	
	11e	LSMF	11e	LSMF	11e	LSMF	11e	LSMF
AC_VO(ms)	0.034	0.036	0.066	0.092	0.16	0.18	0.25	0.24
AC_VI(ms)	0.087	0.14	0.091	0.17	0.24	0.25	0.40	0.47
AC_BE(ms)	0.13	0.15	0.22	0.23	0.63	0.59	2.38	2.28
AC_BK(ms)	0.14	0.16	0.44	0.27	1.04	0.90	10.1	3.44

¹The dropped packets in underloaded cases and the AC_BK in overloaded cases are too few to be shown here.

²The retry limit can be made larger for fewer dropped packets, but the delay will increase.

³The mean value is collected on a 600-second simulation. Note in the underloaded scenario, the mean value might have some discrepancy when the simulation seed value varies.

In the overloaded cases, the delay for the AC_VI and AC_BK increases linearly due to the accumulation of backlogged packets. The delay for AC_BE is not an important performance indicator, so we choose the delay for AC_VO as the key performance target. The delay here is counted from the time the packet is generated into the station until it has been being received successfully. This includes the time on transmissions and retransmissions, as their influence cannot be ignored in the overloaded cases. The delay distribution of AC_VO packets collected in a 30-second span in the overloaded 802.11e and LSMF scenarios is shown in Figure 6. Values on the y-axis represent the number of packets whose delay is below the coordinated x-axis value. So the end point on the y-axis also indicates the total number of successfully transmitted packets. The start point indicating $40\mu\text{s}$ on the x-axis represents packets being transmitted immediately when the channel is idle. $40\mu\text{s}$ is simply the time of transmission. The solid lines represent packet numbers in 802.11e and dotted lines are for those in LSMF. Lines of the same colour are simulated with the same number of stations.

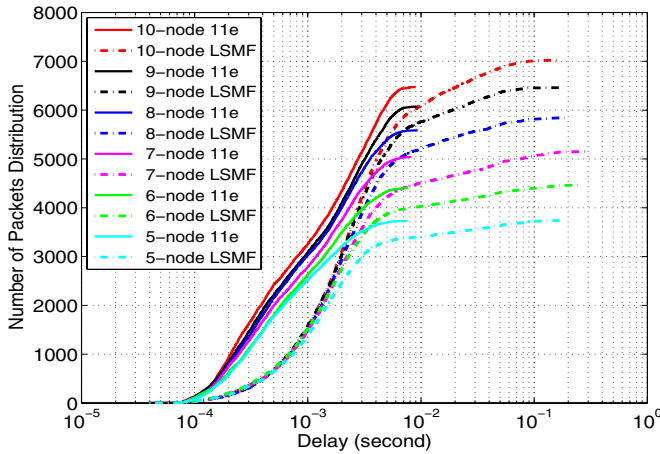


Fig. 6. AC_VO delay distribution in 802.11e and in LSMF

TABLE III
DISTRIBUTION OF PACKETS NUMBER WITH 10-NODE

Delay	Number of Packets				
	< 100 μs ≥ 40 μs	< 1ms ≥ 100 μs	< 10ms ≥ 1ms	< 161ms ≥ 10ms	total
11e	133	3123	3220	0	6476
LSMF	39	1516	4535	934	7024

All the six overloaded scenarios show that 802.11e has more packets with a delay less than 1ms, compared to LSMF, which has more packets transmitted between 1ms and 10ms. LSMF has a heavy tailed delay beyond 10ms. To further demonstrate the delay distribution, Table III gives the exact packet numbers within the indicated delay range in the 10-node transmitting network. The reason for longer delay is that there are more low-priority packets transmitted in LSMF and they delay the VOIP packets, as the newly arriving VOIP packet will

not be processed until the current transmission is finished; while in 802.11e, it will join the competition straightaway. AC_VO delay is a trade-off for fewer collisions. In 802.11e, the dropped packets are about four times as many as in the LSMF solution due to collisions. Because the scheduler in the LSMF model is work-conserving⁴ (in this configuration), this is consistent: an increase in throughput must have an impact on the delay.

The distribution in LSMF is heavy tailed, indicating that more packets are suffering longer delay. However, in Figure 6 and Figure 4, we can see both that the throughput is maintained, and that this may have more benefits in allowing at least some traffic to pass in all categories.

V. FURTHER DISCUSSIONS ABOUT LSMF

We have proved that the LSMF structure is comparable to 802.11e when the network is not overloaded and superior to the MSM structure in 802.11e when the network is overloaded. In general, this idea can enhance radio efficiency (utilisation of the medium) as well as providing effective QoS for different traffic types. Furthermore, the LSMF has two more advantages: firstly, the local scheduler itself is flexible and can be implemented according to different requirements; secondly, the LSMF structure is compatible with other QoS schemes which are designed for single stream per node case.

A. Other implementation examples

The local scheduler in the LSMF architecture can be implemented to use any typical QoS schemes. The one we implemented in Section IV is to emulate the behaviour of 802.11e as much as possible. In this subsection, we give two other examples to show that the LSMF architecture can be easily extended to other implementations. The first is a WFQ scheduler we implemented; the second is from a published paper.

a) *WFQ*: The WFQ scheduler is designed to allocate the four ACs' load to the MAC layer at a ratio of 8:4:2:1. At the start of the WFQ execution, each flow is given a quota, which follows the 8:4:2:1 proportion. If one traffic flow is empty, the bandwidth is automatically available to other flows. But the empty flow can build up its credit up to five times its quota. To show the WFQ implementation works well in the LSMF architecture, we have the following simulation:

The simulation is run in a 10-node network at 54Mbps PHY speed. The AC_VO is in an off-on traffic generation. The off and on states alternate every 20 seconds. The other three flows are always on. The packet length of each traffic type is still the same as those in Section IV, except that the AC_VO is made as long as 320 bytes. Figure 7 shows the throughput following a 8:4:2:1 shape⁵. Overall, the throughput of a high priority traffic class is always kept twice as high as the next lower priority. At the beginning of AC_VO's on-period, it has

⁴The scan mentioned in Section III-B can be more aggressive, e.g. scanning every time when the low priority packets resume the backoff timer as well as encounter a transmission failure.

⁵If the dropped packets are included, the statistics will be strictly 8:4:2:1.

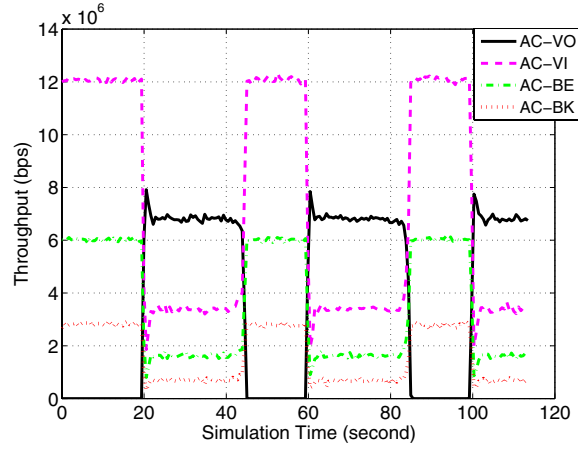


Fig. 7. Throughput of WFQ model

a throughput boost due to the built-up credit from the off-period. After the 20-second on-state, AC_VO still has some throughput lasting for a while. These packets are those queued earlier in the buffer.

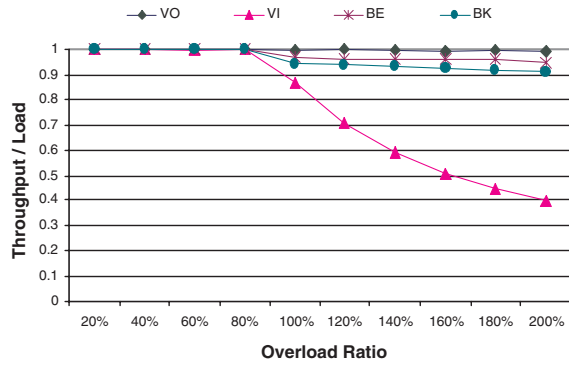


Fig. 8. Throughput/Load of WFQ model with a realistic traffic load

When we apply the traffic setting in Section IV to this WFQ model, the ratio of Throughput/Load is given in Figure 8. With the exception of AC_VI, the remaining bandwidth needed by the other three flows is below their quota. The remainder is available to AC_VI but AC_VI still has too many packets. The throughput of all flows is higher than in the previous two models because there is less contention in the WFQ model and thus fewer dropped packets. The cost is longer delay of AC_VO (e.g. in the 10-node network): the number of packets with delay less than 10 ms is 3852, compared to 6476 in 802.11e.

b) other published model: There is other research using the LSMF structure. For example, the proportional differentiation in [14] has the same node architecture as our LSMF. The implementation for the local scheduler is different from ours; it schedules the packet according to the waiting time of the head-of-line packet. Let $w_i(t)$ denote the time that the head-of-line packet in class i has waited in the queue. The waiting

time parameter $\tilde{w}_i(t)$ at time t is defined as:

$$\tilde{w}_i(t) = \frac{w_i(t)}{\delta_i}$$

where δ_i is the delay differentiation parameter of class i . The scheduler schedules a packet for transmission from the class with the maximum $\tilde{w}_i(t)$. Figure 9 is an example copied from [14].

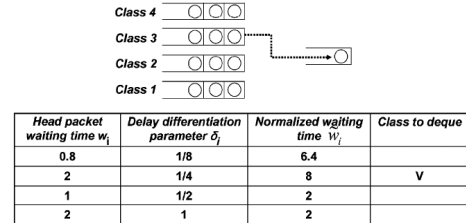


Fig. 9. An example of implementation of LSMF (copy)

Similarly, the waiting time can be replaced by throughput or other criteria according to the operator's requirement.

B. Benefits of using LSMF

802.11e puts all its traffic queues in contention with each other, which intensifies the competition in the distributed wireless network. It is necessary to incorporate a self-checking mechanism in each individual station to control its own transmission decisions, especially for those with aggressive traffic streams. For example, if one user in an ad hoc WLAN opens several streams of multimedia traffic, the other users' applications will be adversely affected. This may lead to aggressive use of the wireless medium by multimedia traffic, as with the *resource hog* problem in computer operating systems.

In the LSMF model, the QoS is taken as a local centralised system at the single station level. In practice, networks are not deliberately operated with excessive traffic, although this may occur from time to time in any system due to transients of local load or retransmissions. A controller around the MAC is very helpful in reducing such transients as well as in preventing malicious or aggressive streams from getting into the WM and reducing the overall QoS. At the same time, it helps to mitigate the competition in distributed wireless networks.

The LSMF structure has solved the problem of handling multiple services internally in a single node. At the network level, the LSMF architecture has several advantages over the MSM in 802.11e in integrating other published proposals to solve the challenges faced by distributed WLANs, e.g. tuning MAC access parameters. Understanding how these schemes work is helpful in assessing the compatibility of LSMF with them.

The relationships among CW sizes, transmission probability τ at a generic timeslot, collision probability, and throughput are proved in the work of Cali [16] and Bianchi [8]. Some published research decreases collision rate and thus improves performance by tuning the CW size. Based on the theory from [8], [19] estimates the number of competing nodes

by observing the idle backoff time. Stations then make the estimation in a distributed manner and set their own CW to the optimal values. [11] also configures the optimal CW size to achieve proportional bandwidth utilisation by different priority classes, but it assumes the number of contending nodes in each class is identified in the beacon period and known to everyone, i.e. it is not fully distributed.

Rather than tuning the CW size, the contention control in [18] [17] is implemented by tuning the rate of dequeuing packets to the MAC - actually it reduces the contention by changing the number of competing nodes. Cali [16] has identified that a balance of bandwidth utilisation between the idle period (backoff time) and collisions can be achieved. Equations of collision numbers and the idle slots are proved in [16]. [17] uses this research result: it counts the number of collisions $E[N_c]$ between two consecutive successful transmissions or the idle backoff slots $E[idle]$ between two transmissions to adjust the packet dequeuing rate. Actually this is consistent with our simulations in Section IV: fewer collisions occurred in LSMF than in 802.11e because of fewer contending entities. The work [17] can be easily extended in the LSMF model: the local scheduler is responsible not only for choosing a packet from all available queues, but also for deciding when to send this packet to the MAC.

All the above published papers [17] [19] [11] [20] assume each node only carries traffic for a single class. With the LSMF model, after the local scheduling, the Q'STA only represents one type of traffic at one time. These schemes can easily be applied to the LSMF model. Moreover, the QoS schemes (DRAFT [12], DWFQ [13], etc) can also be implemented in the MAC of the LSMF model.

The only problem that the LSMF structure has is the possibility of exposing the delay-sensitive traffic to longer delay as it could be held up by the lower priority traffic. When an AC_BK is chosen to represent the Q'STA in contention with other Q'STAs, if, unfortunately, all the others have more competitive traffic, this data packet will occupy the MAC for a relatively longer time. Under these circumstances, the controller should choose the next packet to replace the current selection to avoid delaying transmission of multimedia traffic.

VI. CONCLUSION

This paper has proposed the LSMF structure for the case when multiple flows are present and flowing through every single station. We proved that LSMF improves the medium utilization compared with the MSM structure, which is defined in the 802.11e standard for multiple services. Another advantage of this architecture is that the LSMF has much flexibility and compatibility. The local scheduler in our proposed model can work with any scheduling mechanism, e.g. the weighted fair queuing algorithm. Since each station in the LSMF represents one traffic type at a time at the network level, LSMF is compatible with existing QoS schemes, which normally assume every station only having one priority traffic flow.

ACKNOWLEDGEMENT

The work reported in this paper has formed part of the OSIRIS project within the 3C Research programme of convergent technology research for digital media processing and communications, whose funding and support is gratefully acknowledged. For more information please visit www.3cresearch.co.uk.

The authors would like to thank Dr E J Collins from the Statistics Group in Department of Mathematics in the University of Bristol for his help on the mathematical modelling.

REFERENCES

- [1] IEEE Std 802.11e, November 2005, Amendment 8: Medium Access Control (MAC) Quality of Service (QoS) Enhancements.
- [2] S. Mangold, S. Choi, IEEE 802.11e Wireless LAN for Quality of Service, in Proc. European Wireless'02, Florence, Italy, Feb. 2002.
- [3] S. Choi, J. Prado, S. Shankar N, and S. Mangold, IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation, in Proc. IEEE ICC'03, Anchorage, Alaska, USA, May 2003.
- [4] S. Mangold, S. Choi, Analysis of IEEE 802.11e for QoS Support in Wireless LANs, IEEE Wireless Communications, Dec 2003.
- [5] J. W. Robinson, T. S. Randhawa, Saturation Throughput Analysis of IEEE 802.11e Enhanced Distributed Coordination Function, IEEE Journal on Selected Areas in Communications, Vol.22, No.5, June 2004.
- [6] S. Lu, V. Bharghavan, and R. Srikant, Fair Scheduling in Wireless Packet Networks, IEEE Trans. Networking, pp.473-489, Aug. 1999.
- [7] N. H. Vaidya, P. Bahl, and S. Gupta, Distributed Fair Scheduling in a Wireless Lan, in Proc. ACM International Conference on Mobile Computing and Networking (MOBICOM), Boston, MA, August 2000, pp.167-178.
- [8] G. Bianchi, Performance Analysis of the IEEE 802.11 Distributed Coordination Function, IEEE JSAC, Vol.18, No.3, March 2000.
- [9] E. Zouva and T. Antonakopoulos, CSMA/CA Performance under High Traffic Conditions: Throughput and Delay Analysis, Computer Communications, vol. 25, pp. 313-321, Feb. 2002.
- [10] <http://www.opnet.com>
- [11] C. Hu, H. Kim, J. C. Hou and etc, Provisioning Quality Controlled Medium Access in UltraWideBand-Operated WPANs, INFOCOM06.
- [12] W. Pattara-Atikom, S. Banerjee, and P. Krishnamurthy, Quality of Service Support in IEEE 802.11 Wireless LAN, PhD dissertation, School of Information Sciences and Telecommunications, University of Pittsburgh, March 2005.
- [13] A. Banachs and X. Perez, Distributed Weighted Fair Queuing in 802.11 Wireless LAN, in Proc. IEEE International Conference on Communications (ICC), New York, NY, USA, April 2002, vol.5, pp.3121-3127.
- [14] K. Wang and P. Ramanathan, QoS assurances through class selection and proportional differentiation in wireless networks, IEEE Journal on Selected Areas in Communications, Vol.23, No.23, pp.573-584, March 2005.
- [15] L. Peterson, Computer Networks: A System Approach, Third Edition, 2003, pp.462.
- [16] F. Cali, M. Conti and E. Gregori, Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit, IEEE/ACM Transactions on Networking, Dec. 2000.
- [17] Chunyu Hu and Jennifer Hou, A Novel Approach to Contention Control in IEEE 802.11e-Operated WLANs, IEEE INFOCOM, May 2007.
- [18] H. Kim and J. C. Hou, Improving Protocol Capacity for UDP/TCP Traffic with Model-Based Frame Scheduling in IEEE 802.11-Operated WLANs, IEEE Journal on Selected Areas in Communications, vol.22, No.10, pp. 1987-2003, Dec.2004.
- [19] D. Qiao and K. Shin, Achieving efficient channel utilization and weighted fairness for data communications in IEEE 802.11 WLAN under the DCF, International Workshop on Quality of Service, May 2002.
- [20] Yaling Yang and Robin Kravets, Throughput guarantees for multi-priority traffic in ad hoc networks, Elsevier Ad Hoc Networks, vol.5, No.2, pp.228-253, March 2007.